

Computer Science 138

Course Policies

Spring 2009

IMPORTANT: Please read this document and sign below. Return it to a TA on hours or after class. You will not receive credit for any assignments until we have this on file.

Most CS courses at Brown allow virtually no cooperation or discussion between students. Many students and TAs feel that these policies actually stifle the learning process to some extent, and have spoken out in favor of a more liberal and honor-system-based policy that depends on the maturity of the students to know what work should be their own and what they can share with their peers.

So we're doing things a little differently in CS138, loosening up the policies in the hope of stimulating a better learning environment. Without going outside the basic Academic Code (see Basic Policy, p. 3 of "Tenets of Community Behavior and Academic Code 1992-1993"), we are providing the following set of guidelines.

The basic premise is that you should do your own thinking, your own design, and your own coding. You're allowed to talk to other students about the content of the lectures and of the textbook and about high-level concepts in general. You can answer questions from other students about packages used for assignments, as long as the problem is a narrow one and not one that helps in the problem-solving process at large.

On a general level, what is not allowed is that you let yourself be led by another student to the extent that your task becomes significantly less challenging because of your discussion with him or her. More specifically, you should do your own problem solving, program design and decomposition, and design your own data structures. In conversation with other students, be sure not to venture into design and coding specifics, and especially never sit down to discuss an assignment with someone else before you've analyzed the problem in depth on your own.

For the final project (PuddleStore) and only the final project, you are not only permitted, but encouraged to discuss your high-level design with others in the class. This includes discussion of new features to be added and which algorithms are used. However, you are not permitted to share code. The intent is that you share ideas with others and have a chance to debate their merits. You are expected to understand completely the features you implement.

To be blunt, the most blatant violation that can occur is code-copying, and this absolutely will not be tolerated. We reserve the right to do a "wire-pull test" (i.e., ask you to explain your program) and to use all of the tools at our disposal to compare your code to that of other students (including assignments from years past). In a similar vein, make sure that all of your coursework on the filesystem has the proper permissions so that other students cannot view and potentially copy your work. See `chmod(1)` or ask a consultant for help if you don't know how to go about this. Failure to do this can potentially be viewed as criminal negligence.

We believe that this policy is explicit enough to guide your judgment and that we have not left you many gray areas. If you are ever in doubt about the legality of your actions, be sure to clear them with a TA, even if only after the event has occurred. When we confront a student with a case of suspected violation, an answer of "I didn't know that this was wrong" is not likely to find much sympathy.

Again, note that you are expected always to approach a problem initially on your own and seriously attempt to find a solution. You are honor-bound to preserve your independence of thinking. And remember that the TAs should always be your first resource when you have a question or problem.

TA hours are intended for students to use as a resource for getting help with assignments (both programs and homeworks). It is expected that, before coming to a TA for help, you have made a significant attempt on your own to resolve your problems. For programs, in order to receive help, you must have made a serious

attempt to trace your bug to its source, or at least isolate its occurrence to a few specific scenarios. Simply stating “I have a bug” will result in no help whatsoever from your TA other than the friendly suggestion that you try to debug properly to help isolate the problem. If you think you have found a bug in any of the TA-supplied code, don’t just go crying wolf. Please isolate the bug and provide the TAs with a few explicit scenarios in which it occurs so that we can reproduce and subsequently fix the problem.

That being said, asking TAs for help during times not explicitly designed as their hours is forbidden.

I have read and agree to abide by the above policy.

Name (please print)

Signature

Date

I don't have a CS account